MEMORANDUM REPORT ARBRL-MR-03250

A MULTIPLE LINE PLOTTING ROUTINE

Melinda B. Krummerich

March 1983

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

Destroy this report when it is no longer needed.
Do not return it to the originator.

Additional copies of this report may be obtained from the National Technical Information Service, U. S. Department of Commerce, Springfield, Virginia 22161.

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The use of trade names or manufacturers' names in this report does not constitute indorsement of any commercial product.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER MEMORANDUM REPORT ARBRL-MR-03250	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A MULTIPLE LINE PLOTTING ROUTINE		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) MELINDA B. KRUMMERICH		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U.S. Army Ballistic Research Laboratory ATTN: DRDAR-BLI Aberdeen Proving Ground, MD 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 1L161102AH43
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Armament Research & Development Command U.S. Army Ballistic Research Laboratory (DRDAR-BL) Aberdeen Proving Ground, MD 21005		12. REPORT DATE March 1983 13. NUMBER OF PAGES 42
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer program
BASIC computer language
Microcomputer
Plotting programs

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) jmk

The Multiple Line Plotting Routine is a set of programs designed to produce an annotated plot of several data files. These programs were written for use on the Hewlett-Packard 9845C Desktop Computer. Sample plots and program listings are included.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

I. INTRODUCTION

In analyzing ballistic data, it is often useful to compare similar sets of data by plotting them on one graph. The Multiple Line Plotting Routine is designed to provide a means for producing such graphs.

A method is available on Hewlett-Packard's 9845C Desktop Computer to create and recall plotted lines. The computer uses the MAT PLOT command which translates these lines into numbers stored in a three-dimensional array. The array may be called back to be redrawn on the screen or any peripheral plotter. This produces a presentable plot in a short amount of time.

The routines described in this report are designed to run on a Hewlett-Packard 9845C Desktop Computer with the following options: Mass Storage ROM, I/O ROM, Graphics ROM, Structured Programming ROM and Advanced Programming ROM (AP ROM). These last two are not entirely necessary; without them, the user will need to rewrite any IF-THEN-ELSE statements and construct a subroutine to find the maximum and minimum values in a given array.

II. DATA FORMAT AND SPECIAL LANGUAGE FEATURES

All of the programs are written in Hewlett-Packard's Enhanced BASIC language. The programs are tailored to the data files used by ANPACK, a data analysis package described in Reference 1. The package includes data acquisition programs, routines used to communicate with other computers, and a data reduction and analysis section which uses graphical and numerical techniques to analyze data files. The plotting routines are called from this last section, which displays the menu shown in Figure 1.

```
There are many options available in this part of the
ANALYSIS PACKAGE.  After completion of any task below,
this selection menu will be redisplayed.  Selection of
task 0 returns you to the option list for the SCALED
GRAPHICS, ANALYSIS AND CALIBRATION ROUTINE.

0 - RETURN TO PROGRAM SELECTION LIST
1 - PROCESS A CALIBRATION FILE
2 - RETRIEVE/STORE DATA ON DISC
3 - ADJUST INCREMENT AND NO. OF DATA POINTS TO PROCESS
4 - ADJUST DATA ZERO LEVEL
5 - CONVERT DATA TO ENGINEERING UNITS
6 - SPECTRAL ANALYSIS OF DATA
7 - APPLY DIGITAL FILTER TO DATA
8 - ANNOTATED PLOT OF DATA, WITH DISC STORAGE AND/OR RETRIEVAL
9 - DATA INTEGRATION AND/OR DIFFERENTIATION
10 - MULTIPLE DATA FILE MANIPULATION
11 - MEAN AND STANDARD DEVIATION OF A SET OF DATA ARRAYS
12 - ANNOTATED MULTIPLE DATA FILE PLOTS
```

Figure 1. Analysis and Graphics Routine Menu

---

[1] *J.N. Walbert, Data Acquisition and Analysis Software for Interactive Computing Systems in the Interior Ballistics Division," Ballistic Research Laboratory Memorandum Report, in process.*

The data file itself contains a maximum of 4133 points arranged as shown in Table 1. The files are accessed using Hewlett-Packard's fast binary read and write commands, FREAD and FPRINT. This differs from the standard READ# and PRINT# file commands, which may be used with an internal tape drive if a Mass Storage ROM is not installed.

TABLE 1. DATA FILE STRUCTURE

Location	Contents (Corresponding program variable)
1-32	ASCII file label up to 64 characters long (Label$);
33	Starting point for data - initially 1 (Start);
34	Number of data points in the file (Nwords);
35	Current data increment - initially 1 (Inc);
36	Total number of points in the file (Last_word);
37	Interval between points in milliseconds (Delta_t);
38-4133	Data points (array Data).

As mentioned earlier, the 9845C has the capability to translate a given graphics command into a triplet (vector) of numbers. Several such vectors are stored in an array which may be retrieved at any time to either plot the information or add additional vectors. The plot array can store up to 10922 vectors, most of which contain plotted points. Other vectors include pen color, line type and an end-of-information indicator.

The computer assumes it is plotting to a 100-by-100 point display, scaled from zero to 100 left to right and bottom to top. Normally, the user will plot to a different scale. A plotted point will correspond to a vector (x,y,z) in the array, where x and y are computed as a function of the ratio of computer scaling to user scaling, and z is dependent on the operation performed (MOVE, PLOT or DRAW).

III. THE PROGRAMS

The plot routine consists of four separate programs: a multiple-line plot on one set of axes, a plot of one set of data against another (rather than the data versus time), a double y-axis plot where there are two y-axes with different ranges of values sharing a single time axis, and a driver program to shuttle between the plot routines and the menu displayed in Figure 1.

A. Driver Program. This is the communications link between the different plot programs and the main ANPACK graphics programs. Any of the three routines or the ANPACK graphics menu may be accessed from here.

The Driver prints a menu on the screen, as shown in Figure 2, and the user selects the program he wants to use. The appropriate program is loaded and run automatically.

OPTIONS -

0 - RETURN TO MENU LIST
1 - CREATE MULTIPLE-LINE PLOT
2 - PLOT ONE DATA FILE AGAINST ANOTHER
3 - CREATE DOUBLE Y-AXIS PLOT

Figure 2. Plot Routine Menu

B. Multiple Line Plot. The program allows many data files to be plotted on one set of axes. It is important that the data be similar in magnitude. This is very handy in viewing several files, either to detect trends or to compare the data to a computed mean. There is a program in the Analysis and Graphics Routine which will compute the mean of several data files and the standard deviation from that mean. Plotting the mean plus and minus the standard deviation is helpful, especially if there are many files containing measurements taken from the same instrument.

When the Multiple-Line program is called (selection 1 of the Driver program), it will print

ENTER THE NUMBER OF FILES TO BE PLOTTED.

The user types in a number greater than one. The program then asks

ENTER THE NAME OF FILE 1.

If for some reason the file cannot be read from disc (either it's the wrong type or it isn't on the disc), the program will print an error message:

ERROR ON FILE READ --

ERROR nn IN LINE xxxx

DO YOU WANT A DISC CATALOG?

where nn is the error number and xxxx is the line in the program in which the error occurred. A listing of the names of the files on disc may be printed to the screen, and the user is again requested to enter the file name. If the file can be retrieved, the contents are loaded into array Data, its label printed on the screen and the user is asked

IS THIS THE CORRECT FILE?

If this is the right file, information about the file is relayed to the internal printer. This includes the file name, number of data points, length of the file in milliseconds, maximum and minimum values and the time between points. This process is repeated for each remaining file.

Once all the files have been correctly entered, the user enters the maximum and minimum y-axis values and the times over which the data are to be plotted. As an example, the user enters the names of three data files and the program prints the following table.

TABLE 2. DATA FILE INFORMATION LISTING

FILENAME	NWORDS	XMAX	YMIN	YMAX	DELTA TIME
THTZ:C12	1748	174.7	-.016927	.00779413	.1
THTZ+:C12	1748	174.7	-.0157243	.0164498	.1
THTZ-:C12	1748	174.7	-.023741	.00757101	.1

This tells the user each file has 1748 data points with time between points of .1 millisecond for a total time of 174.7 milliseconds. The program also displays the maximum and minimum data values for each file, so the appropriate x- and y-axis values may be chosen.

After the plot boundaries have been set (start and stop, maximum and minimum y-axis values), the program asks for the labels to be printed along the x and y axes. These labels may use any character on the keyboard (except control characters) up to 32 characters in length. They are normally used to indicate the type of measurement (displacement, pressure, time) and its associated units (m, MPa, ms).

The program now retrieves the data files one at a time and asks the user, for each file,

ENTER THE LINE TYPE (ENTER 0 FOR CHART).

The user then chooses from the ten line types available. If he is not familiar with the line types, the user may, by entering 0, display the chart shown in Figure 3. The patterned line types may be spread out by setting the length to a number greater than one. The program will ask for the pattern

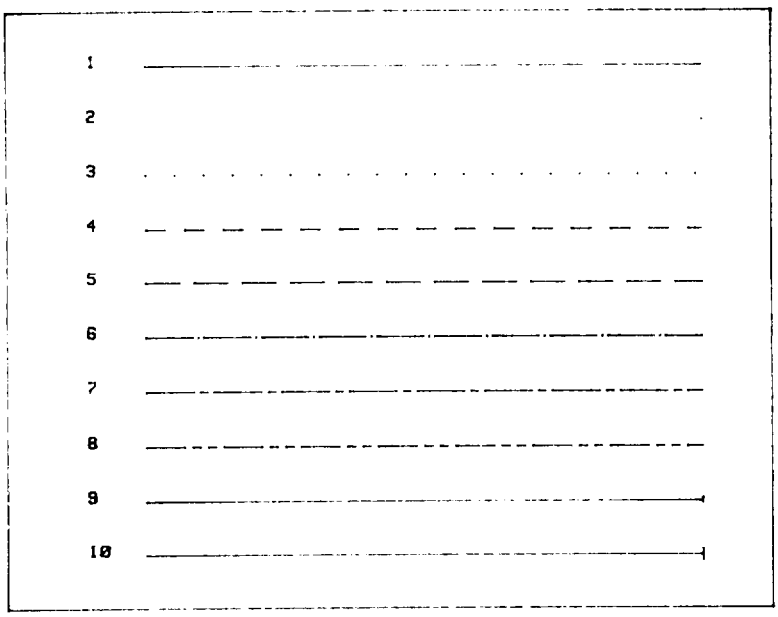

Figure 3. Line Type Chart

length if the user has chosen other than line type 1 (solid line). The flat bed plotter is much more sensitive to the size of the length factor than the screen; therefore, length two may not be noticeably different than length one on the screen, but it is readily apparent on the plotter.

Another thing to keep in mind: The internal printer on the H-P 9845C doesn't print in color; only in black or blue on white. When choosing pen color, use the numbers corresponding to the pen colors in the peripheral plotter. In most cases, these will not correspond to the screen colors.

Figure 4 is a plot of the sample files above, measuring the rotation of a gun tube. The mean file is plotted with line type 1. To achieve contrast, the files showing the deviation from the mean are drawn using the dotted line type 3 of length 2. The frame and labels are always drawn using pen 1; be sure the pen in stable 1 is the color desired for the labels.

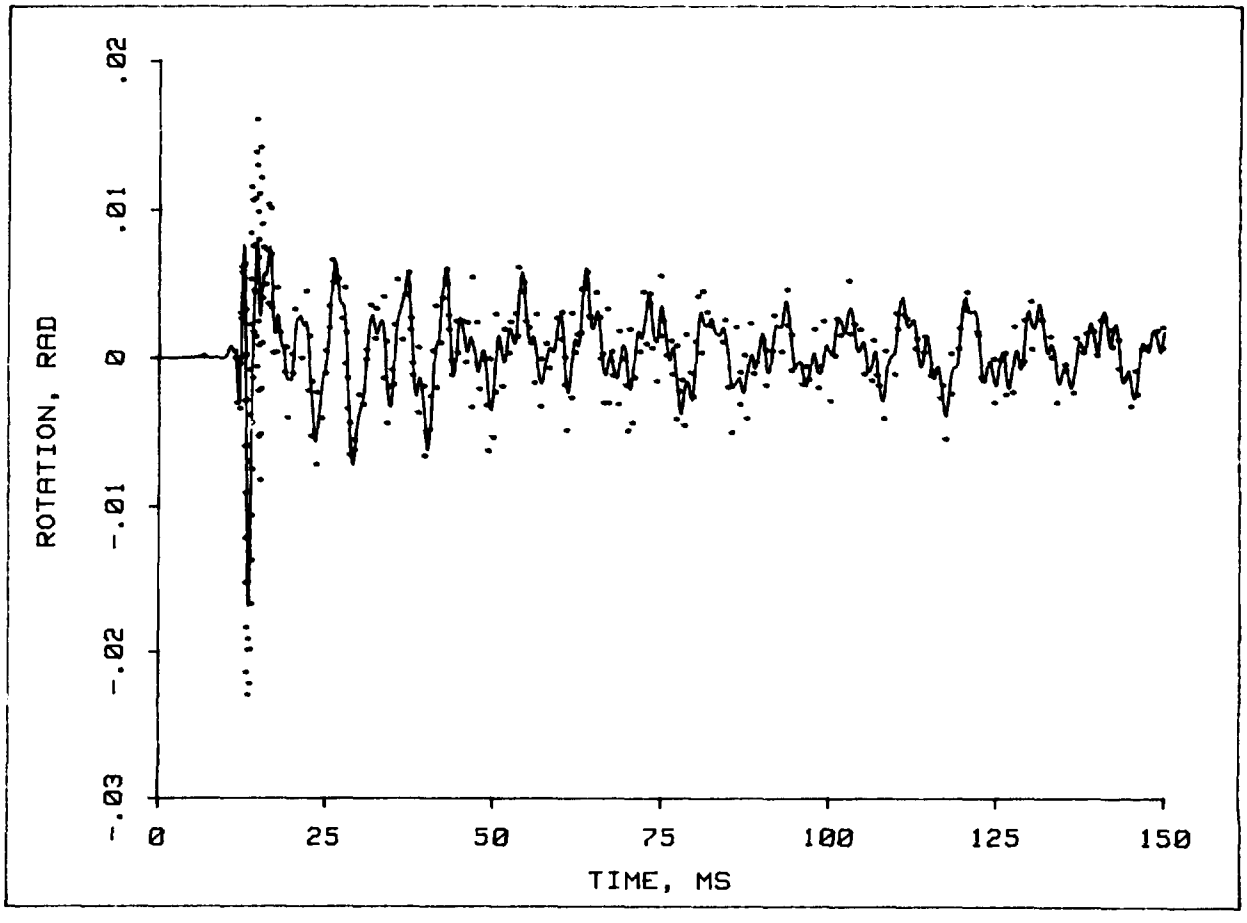

Figure 4. Mean Rotational Displacement Plus and Minus Three Standard Deviations

Figure 5 shows a plot of strain data measured over ten rounds and the mean of those ten rounds. Even though the individual rounds are obscured, the main shape of the envelope is well defined, and a contrasting pen color highlights the mean curve.

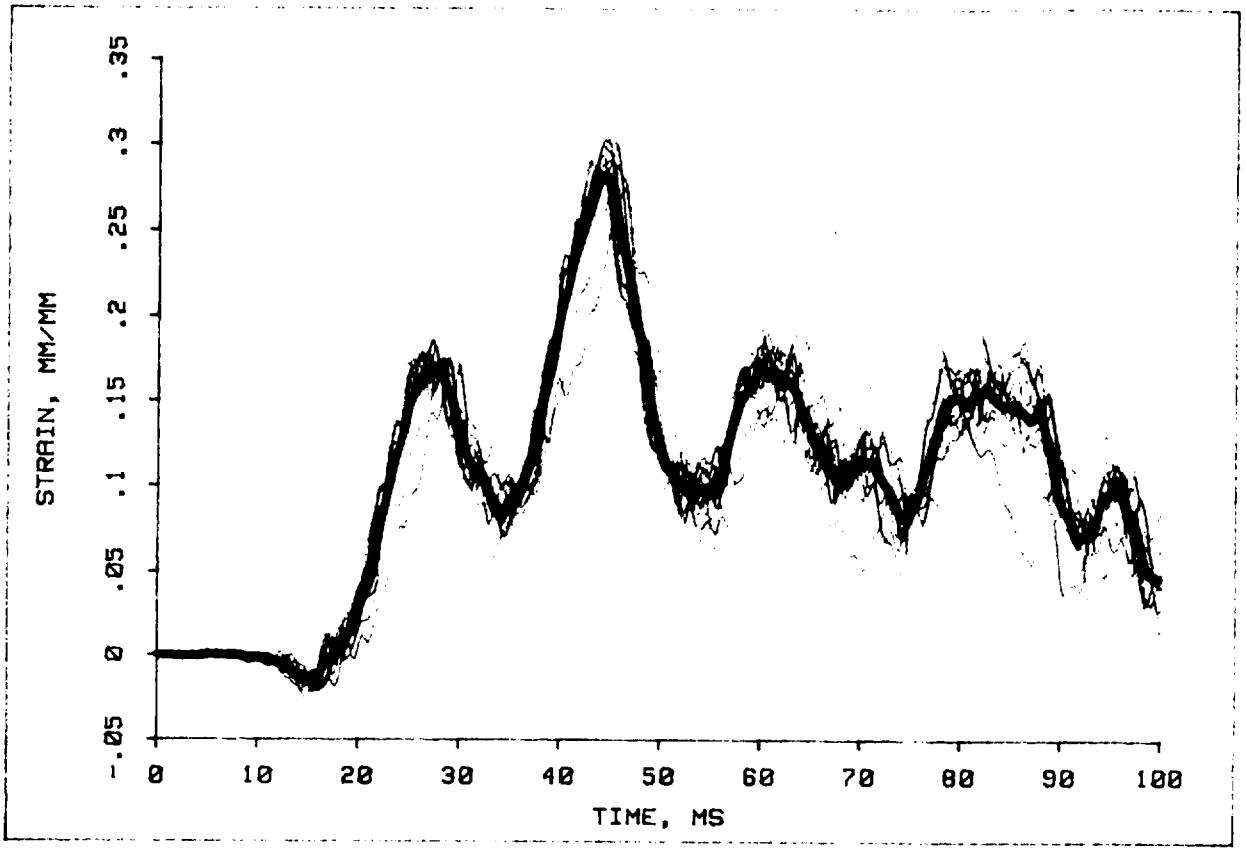

Figure 5. Strain Measured Over Ten Rounds and the Computed Mean

C. Plotting One File Against Another. Normally, all data is plotted versus time. Occasionally, the need to directly compare the data arises. This program will retrieve two data files and plot one against the other over a specified time interval. The files may have different time increments initially, but the program will correct this if the user so desires. It is important that the files be measured over the same time frame; otherwise, the plot is at best useless and at worst, misleading.

The program is called by selecting Option 2 of the Driver program. The words

ENTER THE NAME OF THE X-AXIS FILE

appear at the top of the screen. The user enters the name of the file the dimensions of which will be measured along the x-axis. Once the file is found, the program will print the file label and ask

IS THIS THE CORRECT FILE?

If this is the correct file, the program asks for the y-axis file. If this is not the file the user had in mind, he may answer "N" and the program will prompt him to enter another file name. If an error occurs while retrieving the file, the program will print

```
ERROR ON FILE READ --

     ERROR nn IN LINE xxxx

DO YOU WANT A DISC CATALOG?
```

where nn is the error number and xxxx is the line in which the error occurred. A disc catalog is helpful if you are unsure about the file name or its existence on a particular disc.

In order to simplify the plotting, the files are adjusted to the same length and equal time between points by incrementing and truncating the data. If the time between points is not the same for the two files, the program jumps to a separate section to correct this. The screen is cleared and the following information is printed:

```
TIME BETWEEN POINTS IS NOT THE SAME FOR EACH FILE.
DELTA TIME FOR FILE  NIC1E:C12  IS  .002
DELTA TIME FOR FILE  NIC2E:C12  IS  .001
```

Of course, the actual file names and times will vary with each set of files entered. The user has the option of entering a common time between points or starting over with two different files. If he decides to plot the current files, the program asks for a new time between points. The user should use his best judgement, but a good rule of thumb is to pick the lowest common multiple of the two times. In the example above, a good compromise would be .002 ms between points. The file NIC2E is incremented by two, which will bring the time between points to .002 ms. This incrementing has shortened file NIC2E, and file NIC1E is truncated to this same length. Note that if one time is not a multiple of the other, both files will be incremented to the proper time between points.

The user now must decide how much of the files he wants to plot. He may choose to view some portion of the data (say, the first 25 ms of data) or the entire file. The time is used to determine the first and last points plotted, although these time values are not shown on the final plot.

To illustrate, for the sample files above, the program prints

```
FOR FILE NIC1E:C12:      START  = 0          STOP  =  1.022
FOR FILE NIC2E:C12:      START  = 0          STOP  =  1.022
DO YOU WANT TO PLOT THE FILES:
     0  -  OVER THE ENTIRE INTERVAL
     1  -  OVER PART OF THE INTERVAL
```

If the user answers "0", the program asks for a pen color then begins to plot the curve. Otherwise, it will print

```
ENTER THE START AND STOP TIMES OF THE PLOT INTERVAL
```

The program checks the stop time, making sure it occurs after the start time and that there are enough data points in each file to cover the given interval. If the numbers are not right, the program asks for the times until they are acceptable.

After the plot is finished, the user may get a copy either from the internal printer or the plotter. A sample plot is shown in Figure 6.

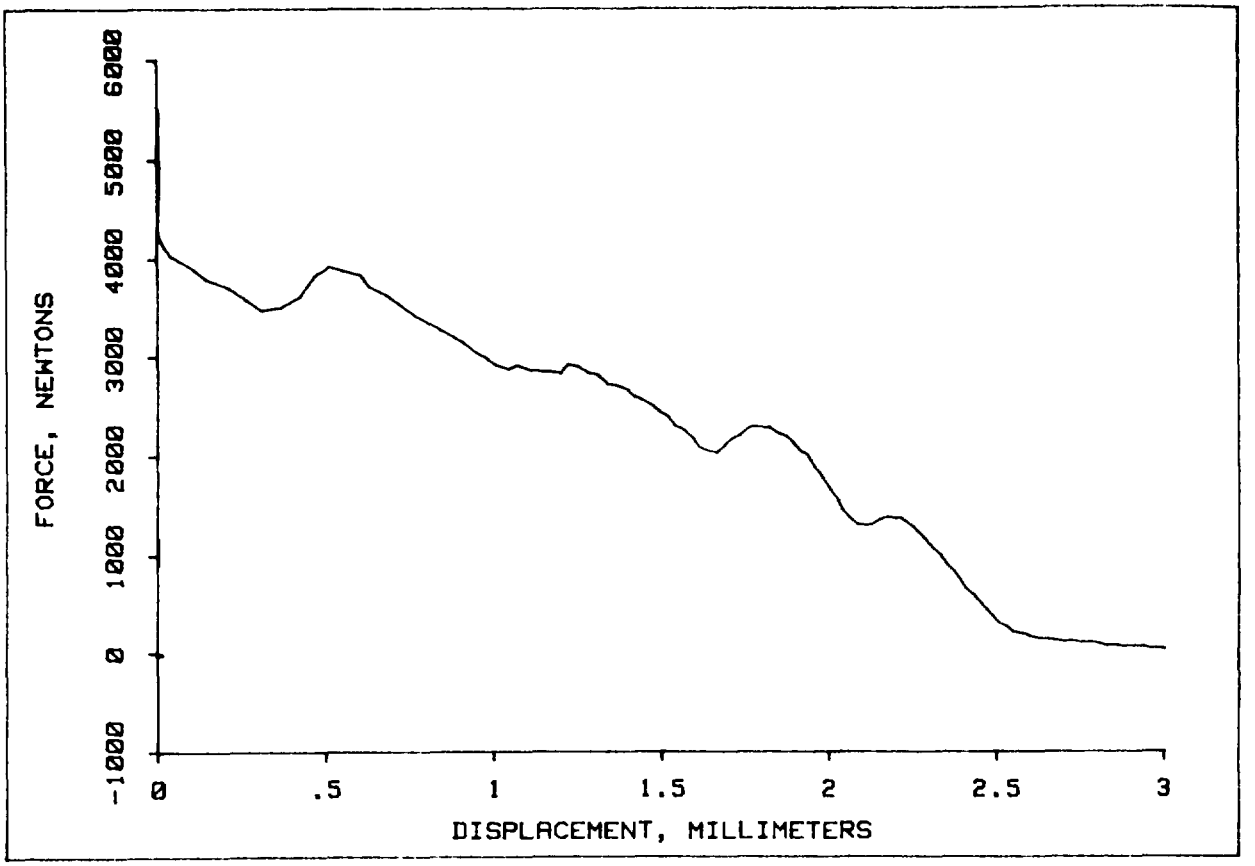

Figure 6. Displacement File vs. Force File

D. Double Y-Axis Plot. This allows for the plotting of dissimilar data over a common time span. It is useful in comparing, say, displacement and acceleration which are related but may not be of the same magnitude. Unfortunately, only two data sets may be plotted this way because the labels become illegible when the plotting area is divided more than twice.

Since the files to be plotted have different magnitudes, each file must be given separate maximum and minimum plot values. These values are variables assigned to the file; i.e., Ymin1 and Ymax1 are used for the first file and Ymin2 and Ymax2 are used for the second file. As a result, it is more efficient to plot each file to a separate array, scaled to its own parameters.

The program begins by asking for the file that will be drawn on the bottom half of the plot. As in the previous programs, if the file cannot be read, an error message appears and the user may obtain a catalog of the disc. The program then searches the file for maximum and minimum values and determines the length of time of the file. The user may change these values by following the program instructions. For the sample files shown in Figure 7, the program would print

```
CURRENT DATA PARAMETERS ARE AS FOLLOWS:
     YMAX = 1293.4
     YMIN = -11388
DO YOU WANT TO CHANGE THE Y-VALUES?
```

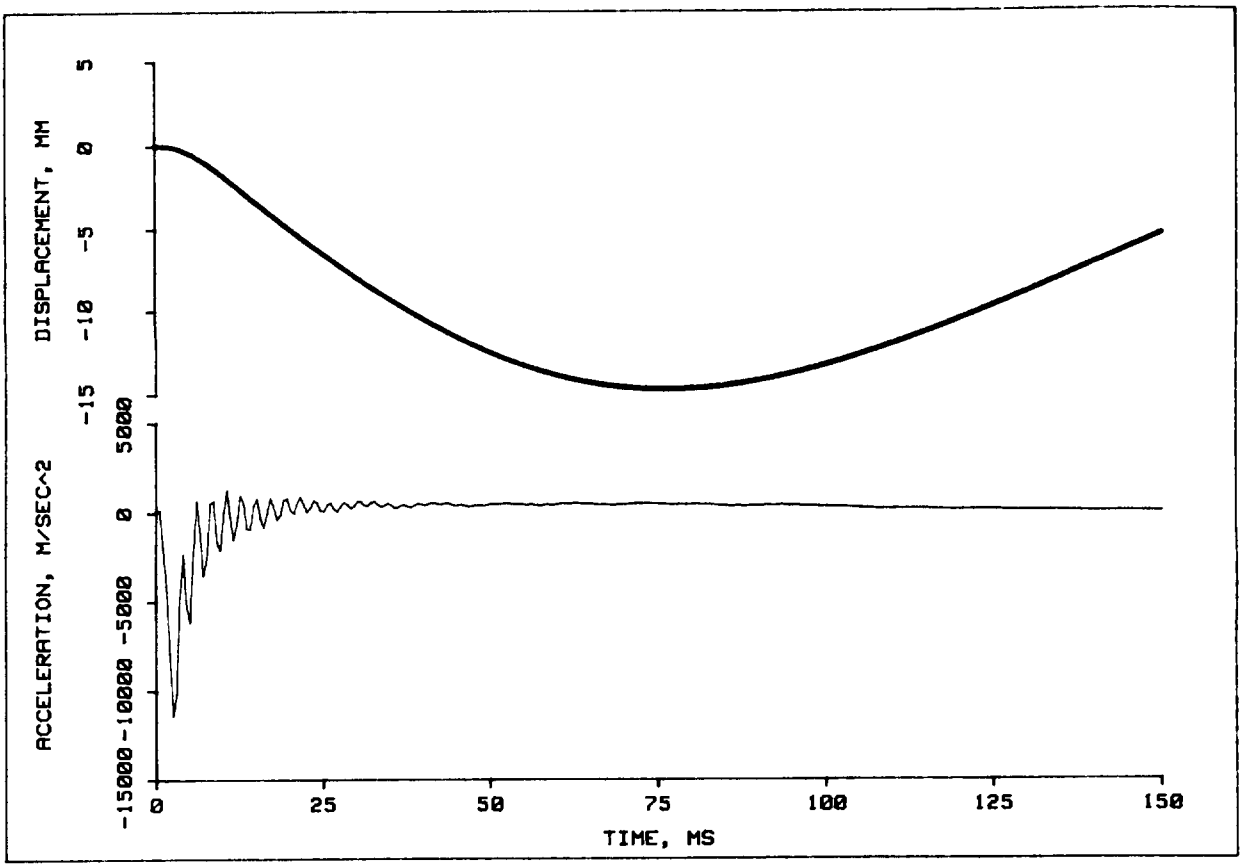

Figure 7. Acceleration and Displacement vs. Time

The user enters "Y" and is then asked to

```
ENTER THE NEW MAXIMUM AND MINIMUM
```

He types in the maximum and minimum, in that order. The program checks the values, assuring itself that the maximum is the larger of the two numbers. These values are then assigned to the variables Ymax1 and Ymin1, respectively, which will be used when the plot is drawn on the peripheral plotter. The same thing is done for the beginning and ending times of the file:

```
CURRENT START AND STOP VALUES:
     START  =  0
     STOP   =  174.5
     DO YOU WANT TO CHANGE THESE VALUES?
```

If the user wants, he may change the times; these times are not assigned separate variables as are the y-axis values above, since the start and stop are the same for both files. The labels are entered for the x-axis and the bottom y-axis, and the first file is plotted to the graphics screen.

The second file is retrieved, and the upper y-axis values are entered as above. Since the start and stop times were entered previously, the program begins to plot the second file.

A copy of the completed plot may be obtained from the internal printer or a peripheral plotter. The user then has the option of replotting the same files to different scale, plotting another pair of files or returning to the Driver program.

IV. SUMMARY AND CONCLUSIONS

The Multiple Line Plotting Routine satisfies the need to produce a high-quality plot of several data files in a short amount of time. It may be used to create plots on paper for reports or on viewgraphs for lectures and presentations.

Because each of the routines is a program, not a subroutine, they may be used independently or together, as presented here. Despite the specialized data sets the programs use, they may be adjusted to accept another type of file structure. They may also be changed to communicate with different plotters by changing one or two lines in each program. These lines will be noted in the program listings in the Appendix.

APPENDIX A

PROGRAM LISTINGS FOR THE MULTIPLE LINE PLOT ROUTINE

APPENDIX A

PROGRAM LISTINGS FOR THE MULTIPLE LINE PLOT ROUTINE

The programs are listed below in the order presented in the report. Figure A-1 shows how the programs interact with each other and the ANPACK Analysis and Graphics Programs.

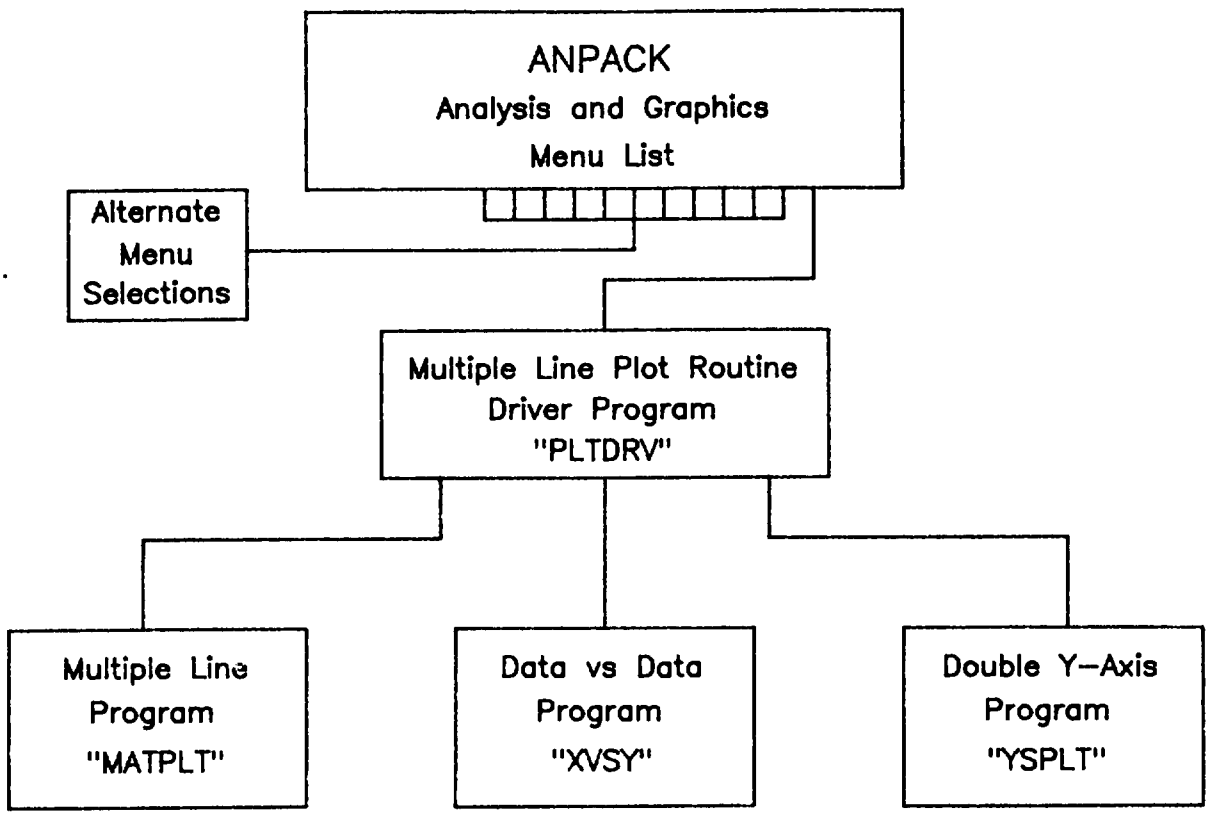

Figure A-1. Program Control Flow Chart

Driver Program

```
10    OPTION BASE 0                                ! FILE "PLTDRV"
20                                                 ! COM variables are used by the
30                                                 ! ANPACK programs.
40                                                 !
50    COM SHORT Start,Stop,Inc,Nrec,Delta_time
60    COM SHORT Data(1:12325),H(1:33),Win(1:33),Wout(1:33)
70    COM SHORT Xmax,Xmin,Ymax,Ymin,Xscale,Yscale
80    COM INTEGER Array_t(23835),Array_b(23940),Array1_t(23835),Array1_b(23940)
90    COM INTEGER First_word,Last_word,Nwords
100   COM Label$[64],Xaxis_label$[32],Yaxis_label$[32],Switch
110   IF GSTAT(1)=7 THEN PLOTTER 7,5 IS OFF
120 Options: PRINT PAGE
130   PRINT "OPTIONS:"
140   PRINT "0 - RETURN TO MENU LIST"
150   PRINT "1 - CREATE MULTIPLE-LINE PLOT"
160   PRINT "2 - PLOT ONE DATA FILE VS ANOTHER"
170   PRINT "3 - CREATE DOUBLE Y-AXIS PLOT"
180   BEEP                                         ! Choose an option, then load the
190   INPUT Option                                 ! appropriate program.
200   PRINT PAGE
210   IF Option=0 THEN LOAD "DRIVER:D12",10        ! All programs are stored on the
220   ON Option GOTO Mult,X_vs_y,Many_y            ! Hewlett-Packard (H-P) Model 7906
230 Mult:  LOAD "MATPLT:D12",10                    ! Disc Drive.  The notation :D12
240 X_vs_y:  LOAD "XVSY:D12",10                    ! refers to the fixed internal
250 Many_y:  LOAD "YSPLT:D12",10                   ! disc.  Data files are stored on
260   END                                          ! removable discs, denoted :C12.
```

Multiple Line Program

```
10    OPTION BASE 1                                  ! FILE "MATPLT"
20    !
30    ! SHORT indicates a short-precision variable (up to six significant
40    ! digits).
50    !
60    ! The variables are:
70    !   Data - array containing the currently plotting data file;
80    !   Plot - array to which files are drawn;
90    !   Files$ - array containing the names of the files to be plotted;
100   !   Numplots - number of files to be plotted;
110   !   Xmin, Xmax - beginning and ending times for a data file;
120   !   Xbegin, Xend - start and stop times for the final plot;
130   !   Ymin, Ymax - upper and lower bounds for the final plot;
140   !   Xunit, Yunit          |>
150   !   Xplus, Yplus          |>
160   !   Type, Length, Pen     |>   Plotting parameters.
170   !   Xlabel$, Ylabel$      |>
180   !
190   SHORT Data(4133),Delta_t,Xmin,Xmax,Xbegin,Xend
200   SHORT Ymin,Ymax,Xunit,Yunit,Xplus,Yplus
210   SHORT Plot(10922,3)
220   INTEGER First_word,Last_word,Nwords
230   INTEGER Numplots,File,Type,Length,Pen
240   DIM Files$(15)[10],Xlabel$[32],Ylabel$[32],Label$[64]
250 Get_files: PRINT PAGE
260   PRINT "ENTER THE NUMBER OF ARRAYS TO BE PLOTTED:"
270   BEEP
280   INPUT Numplots
290   IF Numplots<=1 THEN
300     PRINT LIN(1);"YOU MUST SPECIFY MORE THAN ONE FILE --"
310     BEEP                                         ! The names of the files to be
320     GOTO Get_files                               ! plotted are entered in response
330   END IF                                         ! to the program's questions.
340   REDIM Files$(Numplots)
350   FOR File=1 TO Numplots
360 Get_data: PRINT PAGE,"ENTER THE NAME OF FILE "&VAL$(File)&":"
370     BEEP
380     INPUT Filename$
390     Filename$=Filename$&":C12"
400     ON ERROR GOTO File_error
410     FREAD Filename$,Data(*)
420     OFF ERROR
430     ENTER Data(1) USING "#,64A";Label$
440     PRINT LIN(2);TAB(8);Label$,LIN(1)    ! Once the correct files are
450     PRINT "IS THIS THE CORRECT FILE?"    ! found, the names are placed in
460     BEEP                                 ! array Files$.  These names will
470     INPUT A$                             ! be used later to call the files,
480     IF A$="N" THEN Get_data              ! one at a time, to be plotted.
490     Files$(File)=Filename$               ! Information is printed for each
500     Delta_t=Data(37)                     ! file: Name, Length, Total time
510     First=38                             ! elapsed, Minimum and Maximum
```

```
520     Last=ROW(Data)                          ! values and Time between points.
530     Nwords=Last-37                          ! Lines 560 and 570 use the AP ROM
540     Xmax=Delta_t*(Nwords-1)                 ! to find the maximum and minimum
550     Xmin=0                                  ! values in each file.  If the ROM
560     MAT SEARCH Data,MAX;Ymax,First          ! is not installed, a subroutine
570     MAT SEARCH Data,MIN;Ymin,First          ! to find these values in an array
580     PRINTER IS 0                            ! is required.
590     IF File=1 THEN PRINT USING Header
600     PRINT USING Im;Filename$,Nwords,Xmax,Ymin,Ymax,Delta_t
610     IF File=Numplots THEN PRINT LIN(3)
620     PRINTER IS 16
630   NEXT File
640   PRINT PAGE
650 Y_in: PRINT "ENTER THE MAXIMUM AND MINIMUM Y-AXIS VALUES:"
660   BEEP
670   INPUT Ymax,Ymin                           ! Plotting parameters are entered.
680   IF Ymax<=Ymin THEN                        ! Each file is plotted to the same
690     PRINT "YMAX <= YMIN -- TRY AGAIN"       ! set of axes, but they may be
700     GOTO Y_in                               ! distinguished by using different
710   END IF                                    ! line types and/or pen colors.
720 X_in: PRINT "ENTER THE START AND STOP TIMES:"
730   BEEP
740   INPUT Xbegin,Xend                         ! Lines 810 and 880 use LINPUT
750   IF Xend<=Xbegin THEN                      ! instead of INPUT to allow the
760     PRINT "STOP <= START -- TRY AGAIN"      ! labels to contain punctuation
770     GOTO X_in                               ! marks.
780   END IF
790   PRINT PAGE,"ENTER THE X-AXIS LABEL (32 CHARACTER MAX):"
800   BEEP
810   LINPUT Xlabel$
820   PRINT "ENTER THE INTERVAL BETWEEN TICK MARKS:"
830   PRINT START=";Xbegin,"STOP=";Xend
840   BEEP
850   INPUT Xunit
860   PRINT "ENTER THE Y-AXIS LABEL (32 CHARACTER MAX):"
870   BEEP
880   LINPUT Ylabel$
890   PRINT "ENTER THE INTERVAL BETWEEN TICK MARKS:"
900   PRINT "YMIN=";Ymin,"YMAX=";Ymax
910   BEEP
920   INPUT Yunit
930   PRINT PAGE
940   PLOTTER IS "GRAPHICS"
950   Pen=1
960   FOR I=1 TO Numplots
970     Filename$=Files$(I)
980     FREAD Filename$,Data(*)
990     Delta_t=Data(37)
1000    Last=ROW(Data)
1010 Line: PRINT "ENTER THE LINE TYPE FOR FILE "&VAL$(I)&":"
1020    PRINT "(ENTER 0 FOR LINE TYPE CHART)"
1030    BEEP
1040    INPUT Type
1050    IF Type<=0 THEN Linechart
```

```
1060     IF Type>10 THEN
1070       PRINT "LINE TYPE NOT VALID --"
1080       GOTO Line
1090     END IF
1100     IF Type>1 THEN
1110       PRINT "ENTER THE PATTERN LENGTH:"
1120       PRINT "(THE LARGER THE LENGTH, THE LONGER THE REPEAT DISTANCE)"
1130       BEEP
1140       INPUT Length
1150     END IF
1160     PRINT PAGE
1170     PRINT "DO YOU WANT TO CHANGE PENS?"
1180     BEEP
1190     INPUT P$
1200     IF P$="Y" THEN
1210       PRINT "ENTER THE PEN NUMBER (1-8):"
1220       BEEP
1230       INPUT Pen
1240     ELSE
1250       IF I=1 THEN
1260         Pen=1
1270       END IF                              ! Preparation for plotting: Xloc
1280     END IF                                ! is initialized to the plot's
1290     PLOTTER 13 IS ON                      ! start time.  Yloc is assigned
1300     Xloc=Xbegin                           ! the first data point in array
1310     Yloc=Data(First)                      ! Data.  Xplus and Yplus define
1320     Xplus=.15*(Xend-Xbegin)               ! the area in which the axes and
1330     Yplus=.15*(Ymax-Ymin)                 ! labels are drawn.
1340     PRINT PAGE
1350     GRAPHICS
1360     EXIT ALPHA
1370     SCALE Xbegin-Xplus,Xend+.5*Xplus,Ymin-Yplus,Ymax+.5*Yplus
1380     CLIP Xbegin,Xend,Ymin,Ymax
1390     IF I=1 THEN PLOTTER IS Plot(*)        ! Here is where the plotting takes
1400     PLOTTER Plot(*) IS ON                 ! place.  The pen is positioned at
1410     LINE TYPE Type,Length                 ! the initial values of Xloc and
1420     PEN Pen                               ! Yloc.  The program steps through
1430     MOVE Xloc,Yloc                        ! the array and determines if a
1440     FOR J=First TO Last                   ! point is in the area of interest.
1450       Xloc=Xloc+Delta_t                   ! If so, it is plotted.  If it
1460       IF Xloc<Xbegin THEN Nextj           ! comes before the event, the time
1470       IF Xloc>Xend THEN Cont              ! is increased by Delta_t and the
1480       Yloc=Data(J)                        ! next point is fetched.  If the
1490       PLOT Xloc,Yloc,-1                   ! point occurs after time Xend or
1500   Nextj: NEXT J                           ! the end of the file is reached,
1510   Cont: PENUP                             ! the program exits the loop and
1520     PLOTTER Plot(*) IS OFF                ! gets the next file.
1530     PRINT PAGE
1540     ALPHA
1550     EXIT GRAPHICS
1560   NEXT I
1570   EXIT ALPHA
1580   GRAPHICS                                ! Labels are drawn in after all
1590   UNCLIP                                  ! files have been plotted.  Note
```

```
1600 Labels: LINE TYPE 1                          ! they are not part of the plot
1610  PEN 1                                       ! array, and must be redrawn on
1620  FRAME                                       ! the flat bed plotter.
1630  CLIP Xbegin,Xend,Ymin-.15*Yplus,Ymin
1640  LAXES Xunit,Yunit,Xbegin-10*Xunit,Ymin,-1,-1
1650  UNCLIP
1660  Xcenter=(Xend-Xbegin)/2+Xbegin
1670  Ycenter=Ymin-.75*Yplus
1680  MOVE Xcenter,Ycenter
1690  LORG 5
1700  LABEL Xlabel$
1710  CLIP Xbegin-.15*Xplus,Xbegin,Ymin,Ymax
1720  LAXES Xunit,Yunit,Xbegin,Ymin-10*Yunit,-1,-1
1730  UNCLIP
1740  Xcenter=Xbegin-.75*Xplus
1750  Ycenter=(Ymax-Ymin)/2+Ymin
1760  MOVE Xcenter,Ycenter
1770  LDIR PI/2
1780  LABEL Ylabel$
1790  LDIR 0
1800  PRINT PAGE
1810  ALPHA
1820  EXIT GRAPHICS
1830  PRINT "PRESS CONT WHEN READY TO TYPE LABELS ONTO THE PLOT."
1840  PRINT "WHEN YOU ARE FINISHED, PRESS CONT AGAIN."
1850  PAUSE
1860  EXIT ALPHA
1870  GRAPHICS                                    ! Additional information may be
1880  IF Device=0 THEN LIMIT 5,175,5,145          ! printed on the plot.  This must
1890  CSIZE 3.3                                   ! be re-entered on the peripheral
1900  LETTER                                      ! plotter.
1910  PRINT PAGE
1920  ALPHA
1930  EXIT GRAPHICS
1940  IF Device=1 THEN
1950    PLOTTER 7,5 IS OFF
1960    GOTO Plotter
1970  END IF
1980  PRINT PAGE,"ENTER THE NUMBER OF COPIES DESIRED:"
1990  BEEP
2000  INPUT N
2010  IF N>0 THEN                                 ! A hard copy of the plot may be
2020    PRINT PAGE                                ! dumped to the internal printer or
2030    GRAPHICS                                  ! a peripheral plotter (here, an
2040    EXIT ALPHA                                ! H-P 9872C eight pen plotter).
2050    PRINTER IS 0
2060    FOR I=1 TO N
2070      DUMP GRAPHICS
2080      PRINT LIN(5)
2090    NEXT I
2100    PRINTER IS 16
2110    PRINT PAGE
2120    ALPHA
2130    EXIT GRAPHICS
```

```
2140  END IF
2150 Plotter: PRINT PAGE,"DO YOU WANT A COPY ON THE FLAT BED PLOTTER?"
2160  BEEP
2170  INPUT A$
2180  IF A$="Y" THEN
2190    PRINT "RESET THE FLAT BED PLOTTER. PRESS CONT WHEN READY."
2200    PAUSE
2210    Device=1                                  ! Line 2230 may be changed to
2220    PLOTTER 13 IS OFF                         ! specify another plotter.
2230    PLOTTER IS 7,5,"HPGL",1,8
2240    SCALE Xbegin-Xplus,Xend+Xplus/2,Ymin-Yplus,Ymax+Yplus/2
2250    CLIP Xbegin,Xend,Ymin,Ymax
2260    MAT PLOT Plot                             ! MAT PLOT redraws the data files,
2270    UNCLIP                                    ! then the program returns to draw
2280    GOTO Labels                               ! the labels and the axes.
2290  END IF
2300  PRINT PAGE,"DO YOU WANT TO CHANGE ANY PLOT PARAMETERS?"
2310  BEEP
2320  INPUT A$
2330  IF A$="Y" THEN Y_in
2340  PRINT PAGE,"DO YOU WANT TO PLOT ANOTHER SET OF FILES?"
2350  BEEP
2360  INPUT A$
2370  IF A$="Y" THEN 190
2380  LOAD "PLTDRV:D12",10
2390 File_error:                                ! Error recovery - ERRM$ is a
2400  PRINT "ERROR ON FILE READ --",LIN(1)      ! built-in function which prints
2410  PRINT LIN(1),TAB(5),ERRM$,LIN(1)          ! the error number and where it
2420  PRINT "DO YOU WANT A DISC CATALOG?"       ! occured in the program.  A
2430  BEEP                                      ! catalog may help determine the
2440  INPUT A$                                  ! existence of a file on the disc.
2450  IF A$="Y" THEN CAT ":C12"
2460  DISP "PRESS CONT WHEN READY TO RE-ENTER FILENAME"
2470  PAUSE
2480  GOTO Get_data
2490 Linechart: IF I>1 THEN PLOTTER Plot(*) IS OFF
2500  PLOTTER IS "GRAPHICS"
2510  GRAPHICS                                  ! Line chart - An aid for finding
2520  EXIT ALPHA                                ! a suitable line type for a file.
2530  SCALE 0,10,-11,0                          ! Each type is displayed with its
2540  LORG 2                                    ! identifying number.
2550  FOR L=1 TO 10
2560    MOVE 1,-L
2570    LINE TYPE 1
2580    LABEL L
2590    LINE TYPE L
2600    PENUP
2610    PLOT 2,-L,0
2620    DRAW 8,-L
2630  NEXT L
2640  LINE TYPE 1
2650  FRAME
2660  PAUSE                                     ! If the line chart is called
2670  GCLEAR                                    ! after files have been plotted,
```

```
2680  PRINT PAGE                                ! the screen is cleared and the
2690  ALPHA                                     ! files redrawn using the MAT
2700  EXIT GRAPHICS                             ! PLOT command.
2710  IF I>1 THEN
2720    SCALE Xbegin-Xplus,Xend+.5*Xplus,Ymin-Yplus,Ymax+.5*Yplus
2730    CLIP Xbegin,Xend,Ymin,Ymax
2740    MAT PLOT Plot
2750  END IF                                    ! Format statements used to print
2760  GOTO Line                                 ! file information.
2770 Header: IMAGE X"FILENAME"5X"NWORDS"5X"XMAX"9X"YMIN"8X"YMAX"5X"DELTA TIME"/
2780 Im: IMAGE 10A,5X,DDDD,4(5XK)
2790  END
```

```
10    OPTION BASE 1                                ! FILE "XVSY"
20    !
30    ! SHORT indicates a short-precision variable (up to six significant
40    ! digits).
50    !
60    ! The variables are:
70    !   Data1 - array containing data for x-variable;
80    !   Data2 - array containing data for y-variable;
90    !   Xmin, Xmax - final plot minimum and maximum for Data1;
100   !   Ymin, Ymax - final plot minimum and maximum for Data2;
110   !   Array_t, Array_b - temporary graphics storage arrays;
120   !   Xunit, Yunit      ç>
130   !   Xplus, Yplus      ç>   Plotting parameters.
140   !   Xlabel$,Ylabel$   ç>
150   !
160   SHORT Data1(4133),Data2(4133),Dt
170   SHORT Xmin,Xmax,Ymin,Ymax,Xbegin,Xend
180   SHORT Xunit,Yunit,Delta_t1,Delta_t2
190   INTEGER Array_t(23835),Array_b(23940)
200   INTEGER Nwords1,Nwords2,Last1,Last2
210   INTEGER Flag,First_word,Last_word
220   DIM Label$[64],Xlabel$[32],Ylabel$[32],File1$[10],File2$[10]
230 File1: PRINT PAGE
240   PRINT "ENTER THE NAME OF THE X-AXIS FILE:"
250   BEEP
260   INPUT File1$                                 ! The files are entered into two
270   Flag=1                                       ! arrays - Data1 and Data2.
280   File1$=File1$&":C12"
290   ON ERROR GOTO File_error
300   FREAD File1$,Data1(*)
310   OFF ERROR
320   ENTER Data1(1) USING "#,64A";Label$
330   PRINT LIN(2),TAB(8);Label$,LIN(1)
340   PRINT "IS THIS THE CORRECT FILE?"
350   BEEP
360   INPUT A$
370   IF A$="N" THEN File1
380 File2: PRINT PAGE,"ENTER THE NAME OF THE Y-AXIS FILE:"
390   BEEP
400   INPUT File2$
410   File2$=File2$&":C12"
420   Flag=2
430   ON ERROR GOTO File_error
440   FREAD File2$,Data2(*)
450   OFF ERROR
460   ENTER Data2(1) USING "#,64A";Label$
470   PRINT LIN(2);TAB(8);Label$,LIN(1)            ! Check to see if the time
480   PRINT "IS THIS THE CORRECT FILE?"            ! between points is the same
490   BEEP                                         ! for both files.  If it isn't,
500   INPUT A$                                     ! the program will try to
510   IF A$="N" THEN File2                         ! correct this.
```

```
520   First_word=38
530   PRINT PAGE
540   IF Data2(37)<>Data1(37) THEN Error_exit
550   Dt=Data1(37)
560 Set_vals: !
570   Last1=ROW(Data1)                          ! The program sets Last_word
580   Last2=ROW(Data2)                          ! to the number of points in
590   IF Last1>Last2 THEN                       ! the shorter of the two files.
600     Last_word=Last2
610   ELSE                                      ! The maximum and minimum of
620     Last_word=Last1                         ! each file is found.  If an
630   END IF                                    ! AP ROM is not installed, a
640   MAT SEARCH Data1,MIN;Xmax,First_word       ! routine to find maximum and
650   MAT SEARCH Data1,MIN;Xmin,First_word       ! minimum values is required.
660   MAT SEARCH Data2,MAX;Ymax,First_word
670   MAT SEARCH Data2,MIN;Ymin,First_word
680   PRINT PAGE
690 Parameters: PRINT "CURRENT DATA SPECIFICATIONS ARE:"
700   PRINT TAB(5);"XMIN = ";Xmin
710   PRINT TAB(5);"XMAX = ";Xmax
720   PRINT "DO YOU WANT TO CHANGE THE X-AXIS VALUES?"
730   BEEP
740   INPUT A$
750   IF A$="Y" THEN
760 X_in: PRINT "ENTER THE MAXIMUM AND MINIMUM X-AXIS VALUES:"
770     BEEP
780     INPUT Xmax,Xmin                         ! The user may change the x- or
790     IF Xmin>=Xmax THEN                      ! y-axis values.
800       PRINT "XMAX <= XMIN -- TRY AGAIN"
810       GOTO X_in
820     END IF
830   END IF
840   PRINT PAGE,"CURRENT DATA SPECIFICATIONS ARE:"
850   PRINT "YMIN = ";Ymin,"YMAX = ";Ymax
860   PRINT "DO YOU WANT TO CHANGE THESE VALUES?"
870   BEEP
880   INPUT A$
890   IF A$="Y" THEN
900 Y_in: PRINT "ENTER THE MAXIMUM AND MINIMUM Y-AXIS VALUES:"
910     BEEP
920     INPUT Ymax,Ymin
930     IF Ymax<=Ymin THEN
940       PRINT "YMAX <= YMIN -- TRY AGAIN"
950       GOTO Y_in                             ! The LINPUT command is used in
960     END IF                                  ! lines 1010 and 1080 so the
970   END IF                                    ! labels may use punctuation
980   PRINT PAGE                                ! marks.
990   PRINT "ENTER THE X-AXIS LABEL (32 CHARACTER MAX):"
1000  BEEP
1010  LINPUT Xlabel$
1020  PRINT "ENTER THE INTERVAL BETWEEN TICK MARKS:"
1030  PRINT "XMIN=";Xmin,"XMAX=";Xmax
1040  BEEP
1050  INPUT Xunit
```

```
1060  PRINT "ENTER THE Y-AXIS LABEL (32 CHARACTER MAX):"
1070  BEEP
1080  LINPUT Ylabel$
1090  PRINT "ENTER THE INTERVAL BETWEEN TICK MARKS:"
1100  PRINT "YMIN =";Ymin,"YMAX = ";Ymax
1110  BEEP
1120  INPUT Yunit
1130  PRINT PAGE
1140  PRINT "FOR FILE "&File2$&":  START = 0    STOP = ";Dt*(Last1-First_word)
1150  PRINT LIN(1),"FOR FILE "&File2$&":   START = ";0;"   STOP = ";Dt*(Last2-
First_word)
1160  PRINT LIN(1),"DO YOU WANT TO PLOT THE FILES:"
1170  PRINT TAB(5),"0 - OVER THE ENTIRE INTERVAL"
1180  PRINT TAB(5),"1 - OVER PART OF THE INTERVAL"
1190  BEEP
1200  INPUT Int                                ! Data files which contain more
1210  IF Int=0 THEN                            ! information than necessary may
1220    Xbegin=0                               ! be viewed over a restricted
1230    Xend=Dt*(Last-1)                       ! time frame.  If the user wants
1240  END IF                                   ! to view all the data, he may.
1250  IF Int=1 THEN
1260    PRINT LIN(2),"ENTER THE START AND STOP TIMES OF THE PLOT INTERVAL:"
1270    BEEP                                   ! The starting and ending
1280    INPUT Xbegin,Xend                      ! points are calculated before
1290    IF Xbegin>=Xend THEN                   ! plotting begins.
1300      PRINT "STOP <= START -- TRY AGAIN"
1310      GOTO 1260
1320    END IF
1330    Last=Xend/Dt+First-1
1340    IF Xbegin>0 THEN First=Xbegin/Dt+First-1
1350  END IF
1360  PRINT PAGE,"ENTER THE PEN COLOR (1-8):"
1370  BEEP                                     ! The plot could be highlighted
1380  INPUT Pen                                ! by drawing it in a color other
1390  IF (Pen<1) OR (Pen>8) THEN 1360          ! than black.  NOTE - the pen
1400 Plot_data:                                ! numbers of the screen are not
1410  PLOTTER IS "GRAPHICS"                    ! necessarily the same as those
1420  Xplus=.15*(Xmax-Xmin)                    ! for the plotter!
1430  Yplus=.15*(Ymax-Ymin)
1440  GRAPHICS
1450  EXIT ALPHA
1460  LINE TYPE 1
1470  PEN Pen
1480  SCALE Xmin-Xplus,Xmax+.5*Xplus,Ymin-Yplus,Ymax+.5*Yplus
1490  CLIP Xmin,Xmax,Ymin,Ymax
1500  MOVE Data1(First),Data2(First)           ! The two files are plotted
1510  FOR J=First TO Last                      ! point by point from the start
1520    PLOT Data1(J),Data2(J),-1              ! and stop computed earlier.
1530  NEXT J
1540  PENUP
1550 Labels: !                                 ! The axes and labels are always
1560  PEN 1                                    ! written with pen 1, usually a
1570  UNCLIP                                   ! fine line black pen.
1580  FRAME
```

```
1590  CLIP Xmin,Xmax,Ymin-.25*Yplus,Ymin
1600  LAXES Xunit,Yunit,Xmin-10*Xunit,Ymin,-1,-1
1610  UNCLIP
1620  Xcenter=(Xmax-Xmin)/2+Xmin
1630  Ycenter=Ymin-.75*Yplus
1640  MOVE Xcenter,Ycenter
1650  LORG 5
1660  LABEL Xlabel$
1670  CLIP Xmin-.35*Xplus,Xmin,Ymin,Ymax
1680  LAXES Xunit,Yunit,Xmin,Ymin-10*Yunit,-1,-1
1690  UNCLIP
1700  Xcenter=Xmin-.75*Xplus
1710  Ycenter=(Ymax-Ymin)/2+Ymin
1720  MOVE Xcenter,Ycenter
1730  LDIR PI/2
1740  LABEL Ylabel$
1750  LDIR 0                                  ! GSTORE saves the contents of
1760  GSTORE Array_t(*)                       ! the graphics screen in temporary
1770  GSTORE Array_b(*),0,227                 ! memory.  The screen is displayed
1780  PRINT PAGE                              ! again when the GLOAD command is
1790  ALPHA                                   ! executed.
1800  EXIT GRAPHICS
1810  PRINT "PRESS CONT WHEN READY TO TYPE LABELS ONTO THE PLOT. "
1820  IF Device=0 THEN
1830    PRINT "IF YOU MAKE A MISTAKE, PRESS SOFT KEY k5 TO RE-ENTER LABELS."
1840  END IF
1850  PRINT "WHEN YOU ARE FINISHED, PRESS CONT AGAIN."
1860  PAUSE
1870  PRINT PAGE
1880  GRAPHICS
1890  EXIT ALPHA
1900  CSIZE 3.3                               ! Additional labels may be
1910  ON KEY #5 GOTO 1920                     ! typed on the plot.  The labels
1920  GLOAD Array_t(*)                        ! are stored in the graphics
1930  GLOAD Array_b(*),0,227                  ! array, but they must be
1940  LETTER                                  ! re-typed on the plot drawn
1950  OFF KEY #5                              ! to the peripheral plotter.
1960  GSTORE Array_t(*)
1970  GSTORE Array_b(*),0,227
1980  ALPHA
1990  EXIT GRAPHICS
2000  IF Device=0 THEN
2010    PRINT PAGE,"ENTER THE NUMBER OF COPIES DESIRED:"
2020    BEEP
2030    INPUT Num                             ! Copies of the plot may be dumped
2040    IF Num>0 THEN                         ! to the internal printer or a
2050      PRINT PAGE                          ! peripheral plotter - in this
2060      GRAPHICS                            ! case, an H-P 9872C eight pen
2070      EXIT ALPHA                          ! plotter.
2080      GLOAD Array_t(*)
2090      GLOAD Array_b(*),0,227
2100      PRINTER IS 0
2110      FOR I=1 TO Num
2120        DUMP GRAPHICS
```

```
2130        PRINT LIN(5)
2140      NEXT I
2150      PRINTER IS 16
2160      ALPHA
2170      EXIT GRAPHICS
2180    END IF
2190 Plotter: PRINT "DO YOU WANT A COPY ON THE FLAT BED PLOTTER?"
2200    BEEP
2210    INPUT A$
2220    IF A$="Y" THEN
2230      PRINT "RESET THE FLAT BED PLOTTER. PRESS CONT WHEN READY."
2240      PAUSE
2250      Device=1                              ! The peripheral plotter is
2260      PLOTTER IS 7,5,"HPGL",1,8             ! activated and plot is re-drawn.
2270      GOTO Plot_data                        ! Line 2260 may be changed to
2280    END IF                                  ! specify another plotter.
2290    PRINT PAGE,"DO YOU WANT TO CHANGE ANY PLOT PARAMETERS?"
2300    BEEP
2310    INPUT A$
2320    IF A$="Y" THEN Parameters
2330    PRINT PAGE,"DO YOU WANT TO PLOT ANOTHER PAIR OF FILES?"
2340    BEEP
2350    INPUT A$
2360    IF A$="Y" THEN 150
2370    LOAD "PLTDRV:D12",150
2380 File_error:                                ! If a file is the wrong type
2390    PRINT "ERROR ON FILE READ --"           ! or cannot be found on the disc,
2400    PRINT LIN(1),TAB(8),ERRM$,LIN(1)        ! the program jumps to this
2410    PRINT "DO YOU WANT A DISC CATALOG?"     ! section.  The error type is
2420    BEEP                                    ! listed and a disc catalog may
2430    INPUT A$                                ! be printed to the screen.
2440    IF A$="Y" THEN CAT ":C12"
2450    DISP "PRESS CONT WHEN READY TO RE-ENTER FILENAME"
2460    PAUSE
2470    ON Flag GOTO File1,File2
2480 Error_exit: !
2490    PRINT PAGE,"TIME BETWEEN POINTS IS NOT THE SAME FOR EACH FILE."
2500    PRINT LIN(1);"DELTA TIME FOR FILE "&File1$&" IS ";Data1(37)
2510    PRINT "DELTA TIME FOR FILE "&File2$&" IS ";Data2(37),LIN(2)
2520    PRINT "DO YOU WANT TO RECONCILE THE TIME BETWEEN POINTS?"
2530    BEEP                                    ! The time difference is resolved
2540    INPUT A$                                ! by entering the desired time
2550    IF A$="N" THEN 2330                     ! between points.  One or both
2560    Delta_t1=Data1(37)                      ! files are incremented to this
2570    Delta_t2=Data2(37)                      ! common time, Dt.
2580    PRINT "ENTER THE DESIRED TIME BETWEEN POINTS:"
2590    BEEP
2600    INPUT Dt                                ! Check Dt; if its's the same as
2610    IF Delta_t1=Dt THEN Fix2                ! either Delta_t1 or Delta_t2,
2620    IF Delta_t2=Dt THEN Times=1             ! fix only one file.  If Dt is
2630 Fix1: Inc=Dt/Delta_t1                      ! not equal to either, fix both
2640    I=K=First_word                          ! files and return to the main
2650    Count=1                                 ! body of the program.
2660 Inc1: I=I+Inc
```

```
2670     IF I<=ROW(Data1) THEN
2680       K=K+1
2690       Data1(K)=Data1(I)
2700       Count=Count+1
2710       GOTO Inci
2720     END IF
2730     REDIM Data1(K)
2740     Nwords1=Count
2750     IF Times=1 THEN Set_vals
2760 Fix2: Inc=Dt/Delta_t2
2770     J=K=First_word
2780     Count=1
2790 Incj: J=J+Inc
2800     IF J<=ROW(Data2) THEN
2810       K=K+1
2820       Data2(K)=Data2(J)
2830       Count=Count+1
2840       GOTO Incj
2850     END IF
2860     REDIM Data2(K)
2870     Nwords2=Count
2880     GOTO Set_vals
2890     END
```

Multiple Y-Axis Program

```
10    OPTION BASE 1                          ! FILE "YSPLT:C"
20    !
30    ! SHORT indicates a short-precision variable (up to six significant
40    ! digits).
50    !
60    ! The variables are:
70    !   Data - array containing the currently plotting data file;
80    !   Plot - array to which files are drawn;
90    !   Xmin, Xmax - beginning and ending times for a data file;
100   !   Xbegin, Xend - start and stop times for the final plot;
110   !   Ymin1, Ymax1 - lower and upper bounds for the first file plotted;
120   !   Ymin2, Ymax2 - lower and upper bounds for the second file plotted;
130   !   Flag - indicates which file is being retrieved or plotted;
140   !   Xunit, Yunit1, Yunit2          |>
150   !   Xplus, Yplus1, Yplus2          |>   Plotting parameters.
160   !   Xlabel$, Ylabel1$, Ylabel2$    |>
170   !
180   SHORT Data(4133),Delta_t,Xmin,Xmax,Ymin1,Ymax1,Xplus,Yplus1,Yplus2
190   SHORT Xbegin,Xend,Xunit,Yunit1,Yunit2,Ymax2,Ymin2
200   SHORT Plot1(10922,3),Plot2(10922,3)
210   INTEGER Nwords,First,Last,Flag
220   DIM Xlabel$[32],Ylabel1$[25],Ylabel2$[25],File1$[10],File2$[10]
230   DIM Label$[64]
240 File1: PRINT PAGE,"ENTER THE NAME OF THE FIRST FILE:"
250   BEEP
260   Flag=1                                 ! Let program know this is the
270   INPUT File1$                           ! first file to be retrieved
280   File1$=File1$&":C12"                   ! and plotted.
290   ON ERROR GOTO File_error               ! If the file cannot be found
300   FREAD File1$,Data(*)                   ! or it cannot be read, go to
310   OFF ERROR                              ! the error recovery section.
320   ENTER Data(1) USING "#,64A";Label$     ! Check the file label to see
330   PRINT LIN(2),TAB(8),Label$,LIN(1)      ! if the right file was read.
340   PRINT "IS THIS THE CORRECT FILE?"      ! If not, go back and try it
350   BEEP                                   ! again.
360   INPUT A$
370   IF A$="N" THEN File1
380 Set_vals: Delta_t=Data(37)               ! Find the number of points in
390   First=38                               ! the file, the number of data
400   Last=ROW(Data)                         ! points and the time between
410   Nwords=Last-37                         ! data points for each of the
420   IF Flag=1 THEN                         ! two files.  Initialize the
430     Xmin=0                               ! start and stop the first time,
440     Xmax=Delta_t*(Nwords-1)              ! but not the second time.
450   END IF                                 ! Lines 460 and 470 require the
460   MAT SEARCH Data,MAX;Ymax,First         ! AP ROM.  A sorting routine
470   MAT SEARCH Data,MIN;Ymin,First         ! must be written if it is not
480   PRINT PAGE                             ! installed.
490   PRINT "CURRENT DATA PARAMETERS ARE AS FOLLOWS:"
500   PRINT TAB(5);"YMAX = ";Ymax
510   PRINT TAB(5);"YMIN = ";Ymin
```

```
520   PRINT "DO YOU WANT TO CHANGE THE Y-VALUES?"
530   BEEP                                      ! The user may change the values
540   INPUT A$                                  ! of the maximum and minimum for
550   IF A$="Y" THEN                            ! each file.
560     PRINT "INPUT NEW MAXIMUM AND MINIMUM:"
570     BEEP
580     INPUT Ymax,Ymin
590     IF Ymax<=Ymin THEN
600       PRINT "YMAX <= YMIN  -- TRY AGAIN"
610       GOTO 560
620     END IF
630   END IF                                    ! Ymax and Ymin are assigned to
640   IF Flag=1 THEN                            ! Ymax1 and Ymin1 or Ymax2 and
650     Ymax1=Ymax                              ! Ymin2, depending on which file
660     Ymin1=Ymin                              ! is being plotted.  Ymax and Ymin
670   END IF                                    ! are used to plot each file to
680   IF Flag=2 THEN                            ! the screen, but the other values
690     Ymax2=Ymax                              ! will be used when drawing the
700     Ymin2=Ymin                              ! data to the peripheral plotter.
710     GOTO Plot_two
720   END IF
730   PRINT "CURRENT START AND STOP VALUES:"
740   PRINT TAB(5),"START =";Xmin
750   PRINT TAB(5),"STOP  =";Xmax
760   PRINT "DO YOU WANT TO CHANGE THESE VALUES?"
770   BEEP
780   INPUT A$
790   IF A$="Y" THEN
800     PRINT "ENTER THE NEW START AND STOP:"
810     BEEP
820     INPUT Xbegin,Xend
830     IF Xbegin>=Xend THEN
840       PRINT "STOP <= START -- TRY AGAIN"
850       GOTO 800                              ! Lines 910, 990 and 1000 use the
860     END IF                                  ! LINPUT command instead of INPUT
870   END IF                                    ! so punctuation characters may be
880   PRINT PAGE                                ! written into the labels.
890   PRINT "ENTER THE X-AXIS LABEL (32 CHARACTER MAX):"
900   BEEP
910   LINPUT Xlabel$
920   PRINT "ENTER THE INTERVAL BETWEEN TICK MARKS:"
930   PRINT "START = ";Xbegin,"STOP = ";Xend
940   BEEP
950   INPUT Xunit
960   PLOTTER IS "GRAPHICS"
970 Plot_two: PRINT "ENTER THE Y-AXIS LABEL (25 CHARACTER MAX):"
980   BEEP
990   IF Flag=1 THEN LINPUT Ylabel1$
1000  IF Flag=2 THEN LINPUT Ylabel2$
1010  PRINT "ENTER THE INTERVAL BETWEEN TICK MARKS:"
1020  PRINT "YMAX = ";Ymax,"YMIN = ";Ymin
1030  BEEP
1040  IF Flag=1 THEN INPUT Yunit1               ! Flag keeps the plot parameters
1050  IF Flag=2 THEN INPUT Yunit2               ! separate for each file.
```

```
1060  PRINT "ENTER THE PEN COLOR (1-8):"
1070  BEEP
1080  INPUT Pen
1090  IF (Pen<1) OR (Pen>8) THEN 1060
1100 Begin:                                  ! Position the pen at the first
1110  Xloc=Xbegin                            ! point in the file.
1120  Yloc=Data(First)
1130  Xplus=.15*(Xend-Xbegin)
1140  IF Flag=1 THEN Yplus1=.15*(Ymax1-Ymin1)
1150  IF Flag=2 THEN Yplus2=.15*(Ymax2-Ymin2)
1160  PRINT PAGE                             ! Scale the graphics screen
1170  GRAPHICS                               ! differently for each file,
1180  EXIT ALPHA                             ! using each file's separate
1190 Scale:                                  ! parameters.
1200  IF Flag=1 THEN SCALE Xbegin-Xplus,Xend+Xplus/2,Ymin1-1.5*Yplus1,1.4*(Ymax1
-Ymin1)+Yplus1/2
1210  IF Flag=2 THEN SCALE Xbegin-Xplus,Xend+Xplus/2,2*(Ymin2-Ymax2)-Yplus2,
Ymax2+Yplus2
1220  CLIP Xbegin,Xend,Ymin,Ymax
1230  IF Flag=1 THEN PLOTTER IS Plot1(*)     ! Draw the first file to the
1240  IF Flag=2 THEN PLOTTER IS Plot2(*)     ! array Plot1, the second to
1250  PEN Pen                                ! array Plot2.
1260  MOVE Xloc,Yloc                         ! Increment the time by Delta_t
1270  FOR I=First TO Last                    ! and see if it lies in the
1280    Xloc=Xloc+Delta_t                    ! interval Xbegin to Xend.  If
1290    IF Xloc<Xbegin THEN Nexti            ! it occurs before Xbegin,
1300    IF Xloc>Xend THEN Cont               ! increment the time to the
1310    Yloc=Data(I)                         ! next point.  If it occurs
1320    PLOT Xloc,Yloc,-1                    ! within the interval, the data
1330 Nexti: NEXT I                           ! point is plotted and the loop
1340 Cont: PENUP                             ! is continued.  If it occurs
1350  IF Flag=2 THEN                         ! after Xend, the loop is exited.
1360    PLOTTER Plot2(*) IS OFF
1370    GOTO Y2                              ! The first or second plot array
1380  END IF                                 ! is turned off (depending on
1390  PLOTTER Plot1(*) IS OFF                ! which file was plotted) and
1400 Label_one: UNCLIP                       ! the appropriate labels drawn.
1410  CSIZE 2.75
1420  PEN 1
1430  CLIP Xbegin,Xend,Ymin1-.15*Yplus1,Ymin1
1440  LAXES Xunit,Yunit1,Xbegin-10*Xunit,Ymin1,-1,-1
1450  UNCLIP
1460  Xcenter=.5*(Xend-Xbegin)+Xbegin
1470  Ycenter=Ymin1-1.125*Yplus1
1480  MOVE Xcenter,Ycenter
1490  LORG 5
1500  LABEL Xlabel$
1510  CLIP Xbegin-.1*Xplus,Xbegin,Ymin1,Ymax1
1520  LAXES Xunit,Yunit1,Xbegin,Ymin1-10*Yunit1,-1,-1
1530  UNCLIP                                 ! After the first set of labels
1540  LDIR PI/2                              ! is drawn to the screen, the
1550  Xcenter=Xbegin-.75*Xplus               ! second file is retrieved and
1560  Ycenter=Ymin1+.5*(Ymax1-Ymin1)         ! plotted.  If the labels are
1570  MOVE Xcenter,Ycenter                   ! being drawn on a peripheral
```

```
1580  LABEL Ylabel1$                              ! plotter, the second file is
1590  IF Device=1 THEN                            ! drawn, then the second set of
1600    UNCLIP                                    ! axes.
1610    SCALE Xbegin-Xplus,Xend+Xplus/2,2*(Ymin2-Ymax2)-Yplus2,Ymax2+Yplus2
1620    CLIP Xbegin,Xend,Ymin2,Ymax2
1630    MAT PLOT Plot2
1640    UNCLIP
1650    GOTO Y2
1660  END IF
1670  WAIT 1000
1680  ALPHA
1690  EXIT GRAPHICS
1700 File2: PRINT "ENTER THE NAME OF THE SECOND FILE:"
1710  BEEP                                        ! After the first file has been
1720  INPUT File2$                                ! drawn to the screen, the second
1730  File2$=File2$&":C12"                        ! file is retrieved.  Flag is
1740  Flag=2                                      ! updated to 2 and the program
1750  ON ERROR GOTO File_error                    ! goes to line Set_vals, where
1760  FREAD File2$,Data(*)                        ! the second plot's parameters
1770  OFF ERROR                                   ! are set.
1780  ENTER Data(1) USING "#,64A";Label$
1790  PRINT LIN(2),TAB(8),Label$,LIN(2),"IS THIS THE CORRECT FILE?"
1800  BEEP
1810  INPUT A$
1820  IF A$="N" THEN File2
1830  GOTO Set_vals
1840 Y2: !
1850  PEN 1
1860  CLIP Xbegin-.35*Xplus,Xbegin,Ymin2,Ymax2
1870  LAXES Xunit,Yunit2,Xbegin,Ymin2-10*Yunit2,-1,-1
1880  UNCLIP
1890  Xcenter=Xbegin-.75*Xplus
1900  Ycenter=Ymin2+.5*(Ymax2-Ymin2)
1910  MOVE Xcenter,Ycenter
1920  LABEL Ylabel2$
1930  FRAME
1940  WAIT 1000
1950  PRINT PAGE
1960  ALPHA
1970  EXIT GRAPHICS
1980  PRINT "PRESS CONT TO TYPE IN LABELS."
1990  CSIZE 2.75
2000  LDIR 0
2010  PAUSE
2020  GRAPHICS
2030  PRINT PAGE
2040  EXIT ALPHA
2050  LETTER
2060  ALPHA
2070  EXIT GRAPHICS
2080  IF Device=1 THEN Flat_bed
2090  PRINT "ENTER THE NUMBER OF COPIES DESIRED:"
2100  BEEP
2110  INPUT Num                                   ! The plot may be dumped to the
```

```
2120  IF Num=0 THEN Flat_bed                    ! machine's internal printer or
2130  PRINT PAGE                                ! to a peripheral plotter-- in
2140  GRAPHICS                                  ! this case, an H-P 9872C eight
2150  EXIT ALPHA                                ! pen plotter.
2160  PLOTTER 13 IS ON
2170  PRINTER IS 0
2180  FOR I=1 TO Num
2190    DUMP GRAPHICS
2200    PRINT LIN(5)
2210  NEXT I
2220  PRINTER IS 16
2230  ALPHA
2240  EXIT GRAPHICS
2250 Flat_bed: IF GSTAT(1)=7 THEN
2260    PLOTTER 7,5 IS OFF
2270    PLOTTER 13 IS ON
2280  END IF
2290  PRINT PAGE,"DO YOU WANT A COPY ON THE FLAT BED PLOTTER?"
2300  BEEP
2310  INPUT A$
2320  IF A$="Y" THEN
2330    Device=1                                ! Line 2340 may be changed to
2340    PLOTTER IS 7,5,"HPGL",1,8               ! specify another plotter.
2350    PLOTTER 13 IS OFF
2360    PRINT PAGE,"RESET THE FLAT BED PLOTTER.  PRESS CONT WHEN READY TO PLOT."
2370    PAUSE
2380    GRAPHICS
2390    PRINT PAGE
2400    EXIT ALPHA
2410    UNCLIP
2420    SCALE Xbegin-Xplus,Xend+Xplus/2,Yminl-1.5*Yplusl,1.4*(Ymaxl-Yminl)+Yplus/2
2430    CLIP Xbegin,Xend,Yminl,Ymaxl
2440    MAT PLOT Plotl
2450    UNCLIP
2460    GOTO Label_one
2470  END IF
2480  PRINT PAGE,"DO YOU WANT TO PLOT ANOTHER PAIR OF FILES?"
2490  BEEP
2500  INPUT A$
2510  IF A$="N" THEN LOAD "PLTDRV:D12",10
2520  GOTO File1
2530 File_error: PRINT PAGE,"ERROR ON FILE READ:",LIN(2),TAB(8),ERRM$
2540  PRINT LIN(1),"DO YOU WANT A DISC CATALOG?"
2550  BEEP
2560  INPUT A$
2570  IF A$="Y" THEN CAT ":C12"
2580  ON Flag GOTO File1,File2
2590  END
```

DISTRIBUTION LIST

No. of Copies	Organization
12	Administrator Defense Technical Info Center ATTN: DTIC-DDA Cameron Station Alexandria, VA 22314
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army BMD Advanced Technology Center ATTN: BMDATC-M, Mr. P. Boyd P.O. Box 1500 Huntsville, AL 35804
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCLDC, T. Shirata 5001 Eisenhower Avenue Alexandria, VA 22333
5	Commander USA ARRADCOM ATTN: DRDAR-TDC, Dr. D. Gyorog DRDAR-LC, Dr. J. T. Frasier DRDAR-SE DRDAR-SA DRDAR-AC Dover, NJ 07801
2	Commander USA ARRADCOM ATTN: DRDAR-TSS Dover, NJ 07801
5	Commander USA ARRADCOM ATTN: DRDAR-LCU, E. Barrieres DRDAR-LCU, R. Davitt DRDAR-LCU-M, D. Robertson DRDAR-LCU-M, M. Weinstock DRDAR-LCA, C. Larson Dover, NJ 07801
4	Commander USA ARRADCOM ATTN: DRDAR-LCA, R. Collett DRDAR-LCA, B. Knutelski DRDAR-LCR-R, E. H. Moore III DRDAR-LCS-D, K. Rubin Dover, NJ 07801
8	Commander USA ARRADCOM ATTN: DRDAR-SCM DRDAR-SCM, Dr. E. Bloore DRDAR-SCM, J. Mulherin DRDAR-SCS, B. Brodman DRDAR-SCS, Dr. T. Hung DRDAR-SC, S. Jacobson DRDAR-SCA, W. Gadomski DRDAR-SCA, E. Malatesta Dover, NJ 07801
4	Commander USA ARRADCOM ATTN: DRDAR-LCR, W. Williver DRDAR-LCA, J. Reuben DRDAR-LCA, S. Bernstein DRDAR-LCN, G. Demitrack Dover, NJ 07801
4	Commander USA ARRADCOM ATTN: DRDAR-LCA, Dr. S. Yim DRDAR-LCA, L. Rosendorf DRDAR-LCA, Dr. S. H. Chu DRDAR-LCW, R. Wrenn Dover, NJ 07801
1	Director USA ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL Watervliet, NY 12189
2	Director USA ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB, Dr. T. Simkins DRDAR-LCB, Dr. J. Zweig Watervliet, NY 12189

DISTRIBUTION LIST

No. of Copies	Organization
2	Commander USA ARRADCOM ATTN: Product Assurance Directorate DRDAR-QA Dover, NJ 07801
1	Commander USA ARRADCOM ATTN: DRDAR-TSE, L. Goldstein Dover, NJ 07801
1	Commander US Army Rock Island Arsenal ATTN: DRDAR-TSE-SW, R. Radkiewicz Rock Island, IL 61299
1	Commander US Army Armament Materiel Readiness Command ATTN: DRDAR-LEP-L Rock Island, IL 61299
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-E 4300 Goodfellow Blvd. St. Louis, MO 63120
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
1	Commander US Army Communications Research and Development Command ATTN: DRSEL-ATDD Fort Monmouth, NJ 07703
1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
3	Commander US Army Harry Diamond Laboratories ATTN: DELHD-I-TR, H.D. Curchak DELHD-I-TR, H. Davis DELHD-S-OE-ES, B. Banner 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Harry Diamond Laboratories ATTN: DELHD-TA-L 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Missile Command ATTN: DRSMI-R Redstone Arsenal, AL 35898
1	Commander US Army Missile Command ATTN: DRSMI-YDL Redstone Arsenal, AL 35898
1	Commander US Army Tank Automotive Command ATTN: DRSTA-TSL Warren, MI 48090
2	Commander US Army Jefferson Proving Ground ATTN: STEJP-TD-O, A. Tilley STEJP-TD-E, J. Toomey Madison, IN 47251
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL, Tech Lib White Sands Missile Range, NM 88002
2	Commander US Army Yuma Proving Ground ATTN: STEYP-MTW, R. Torp G. Stullenbarger Yuma, AZ 85364

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander US Army Research Office ATTN: E. Saibel P.O. Box 12211 Research Triangle Park NC 27709
3	Commander US Army Research Office P.O. Box 12211 ATTN: Technical Director Engineering Division Metallurgy & Materials Division Research Triangle Park, NC 27709
2	Project Manager Nuclear Munitions ATTN: DRCPM-NUC Dover, NJ 07801
2	Project Manager Tank Main Armament Systems ATTN: DRCPM-TMA Dover, NJ 07801
2	Project Manager Division Air Defense Gun ATTN: DRCPM-ADG Dover, NJ 07801
3	Project Manager Cannon Artillery Weapons System ATTN: DRCPM-CAWS Dover, NJ 07801
1	Product Manager for 30mm Ammo. ATTN: DRCPM-AAH-30mm Dover, NJ 07801
2	Product Manager M110E2 Weapon System ATTN: DRCPM-M110E2 Rock Island, IL 61299
2	Commandant US Army Infantry School ATTN: ATSH-CD-CSO-OR Fort Benning, GA 31905
1	Commander Naval Sea Systems Command (SEA-03513) ATTN: L. Pasiuk Washington, DC 20362
1	Commander Naval Explosive Ordnance Disposal Facility ATTN: Lib Div Indian Head, MD 20640
1	Superintendent Naval Postgraduate School ATTN: Dir of Lib Monterey, CA 93940
1	Commander Naval Surface Weapons Center ATTN: G-13, W. D. Ralph Dahlgren, VA 22448
5	Commander Naval Surface Weapons Center ATTN: Code G-33, T. N. Tschirn Code N-43, J. J. Yagla L. Anderson G. Soo Hoo Code TX, Dr. W. G. Soper Dahlgren, VA 22448
2	Commander Naval Weapons Center ATTN: Code 3835 Code 3431, Tech Lib China Lake, CA 93555
2	Commander US Naval Weapons Center ATTN: Code 608, R. Derr Code 4505, C. Thelen China Lake, CA 93555

DISTRIBUTION LIST

No. of Copies	Organization
2	Commander US Army Materials and Mechanics Research Center ATTN: J. Mescall Tech Library Watertown, MA 02172
1	Commander Naval Ordnance Station ATTN: Code 5034, C. Irish, Jr. Indian Head, MD 20640
2	Battelle Pacific Northwest Laboratories ATTN: Dr. F. Simonen Mr. E. M. Patton P.O. Box 999 Richland, WA 99352
2	AFATL Gun and Rocket Division Gun Test Branch AD3246 TEST W/TETFG ATTN: W. Dittrich; DLJM D. Davis; DLDL Eglin AFB, FL 32542
1	AFELM, The Rand Corporation ATTN: Library-D 1700 Main Street Santa Monica, CA 90406
1	Southwest Research Institute ATTN: P. Cox 8500 Culebra Road San Antonio, TX 78228

Aberdeen Proving Ground

Dir, USAMSAA
ATTN: DRXSY-D
DRXSY-MP, H. Cohen
Cdr, USATECOM
ATTN: DRSTE-TO-F
Dir, USACSL, Bldg. E3516, EA
ATTN: DRDAR-CLB-PA
DRDAR-CLN
DRDAR-CLJ-L
Dir, USAMTD
ATTN: D. Dykstra
J. Fasig
S. Walton

USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet, fold as indicated, staple or tape closed, and place in the mail. Your comments will provide us with information for improving future reports.

1. BRL Report Number _____

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

_____

_____

_____

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.) _____

_____

_____

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

_____

_____

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) _____

_____

_____

_____

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: _____

Telephone Number: _____

Organization Address: _____

_____

_____

FOLD HERE

Director
US Army Ballistic Research Laboratory
ATTN: DRDAR-BLA-S
Aberdeen Proving Ground, MD 21005

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO 12062 WASHINGTON, DC
POSTAGE WILL BE PAID BY DEPARTMENT OF THE ARMY

Director
US Army Ballistic Research Laboratory
ATTN: DRDAR-BLA-S
Aberdeen Proving Ground, MD 21005

FOLD HERE